

Bipolar Chip Design for a VLSI Microprocessor

In this paper, a pseudo-custom approach to bipolar VLSI chip design is presented, and a hierarchical structure of logic macros assembled from building blocks is described. A strategy of placing the logic macros along with algorithmically designed PLA structures and ROS with a placement aid, and of wiring the placement with an automatic wiring program, is discussed. The paper also focuses on the implementation of this strategy in terms of technology, chip structure, and chip design methodology. In addition, chip statistics are presented and their implications are discussed.

Introduction

The VLSI microprocessor is made up of four bipolar VLSI chips. There are 15 556 equivalent logic circuits plus a 51 200 (50K)-bit control read-only store (ROS) on these four chips. These chips are contained in a 50 × 50-mm module with 361 pins, of which 267 are I/O connections. This module has the highest bipolar circuit density to date ever achieved by IBM at the module level, with 6.2 circuits per mm² of module substrate area. The chips are square and range in size from 6.88 mm (three chips) to 7.2 mm (one chip) on a side. They each have 289 pads in a square array, of which about 150 are I/O connections. The processor module and a few support modules (local store, oscillator, etc.), comprise the VLSI microprocessor engine card; only the main memory resides off card. Internal circuit delays are 3 ns or less in the four chips.

The achievement of a system of this size and performance on one card is the result of a unique application of bipolar VLSI semiconductor technology and multilayer ceramic (MLC) [1] module technology. The module technology is part of IBM's general development effort. However, the chip development was done specifically for the VLSI microprocessor and represents a departure from bipolar gate-arrays [2] for IBM.

The machine design of the VLSI microprocessor is discussed elsewhere in this issue by Campbell and Tahmouh [3]. The subject of this paper is the design and development

of the chips, and the objectives are to describe why fundamental design decisions were made, how the chips are physically and electrically constructed, and how the design methodology was implemented.

The following sections present the reasons for our choice of a custom design approach, the resulting design strategy, the way that strategy translated into technology and chip structure, and the design methodology required to make it happen. In addition, some chip statistics are given, and their implications are assessed.

Choice of custom design approach

A primary hardware objective of the VLSI microprocessor development, which was begun in 1978, was to contain the processor system on only one card in order to reduce product cost. This implied a level of circuit integration in excess of 4000 circuits per chip. In addition, it was apparent that a conventional gate-array approach would not provide us with sufficient flexibility in the chip design. Microprocessors lend themselves more optimally to a high level of integration if imbedded arrays and customized logic circuits suited to particular applications are used. This provides the chip designer with the ability to more closely relate the physical data flow and control wiring to the logic design of the system. Minimizing delay in critical paths and maximizing noise tolerance on long wire lengths, as well as achieving high density, were considerations in making this choice. The

© Copyright 1982 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

nature and extent of the VLSI microprocessor function to be implemented dictated a four-chip partition with this approach.

Chip design strategy

Development costs and resource limitations made it clear that it was not feasible to accomplish four highly customized VLSI chip designs with manual techniques. A streamlined, efficient methodology was developed that was highly automated. It resulted in pseudo-custom chip designs with much replication of circuits and devices and highly structured placements and wiring. Signal wiring was performed with a wiring program, and an automated placement aid was used.

This approach to chip design is more constrained than the FET macro design approach described in [4], where relatively large macros are more truly custom designed. In our case, a hierarchical structure was used in which logic circuit macros were constructed from a limited number of building blocks having a set of design rules covering the logical, physical, and electrical usage of these blocks. Only a few types are necessary. They are a latch, XOR, unit logic (AND-INVERT), drivers, and receivers. Within these types, some variations exist, such as fan-in on unit logic, push-pull or open collector for drivers, etc. The VLSI microprocessor uses about 60 different building blocks, including all variations.

At the next level in the hierarchy there are four types of macros. They are algorithmically programmed logic arrays (PLAs), ROS, logic macros assembled from building blocks, and stand-alone building blocks.

The PLAs were designed with the aid of a design automation tool [5] which automatically generates physical design data based on a logic designer's input. The 50K-bit ROS was designed manually, but was personalized using the ROS personalization program which is part of the IBM Engineering Design System (EDS) [6] software. The logic macros were assembled into $1 \times N$ -byte-wide or less ($N \leq 9$) structures from the building blocks. Figure 1 illustrates the hierarchical concept for logic macros and the use of data flow to simplify wiring. The two logic macros shown are assembled from logic building blocks, and these assemblies are at the next level in the hierarchy above the level of the building blocks. The chip placement of all the logic macros, PLA macros, and the ROS is the highest level in the hierarchy. Interconnection of shapes and associated checking take place at each level in the hierarchy. Making each building block the same width, regardless of function, and placing its wiring channels and LSTs (logic service terminals) on precisely the same horizontal pitch as every other block result in the vertical data flow from logic macro to logic macro being implemented in short, straight wires.

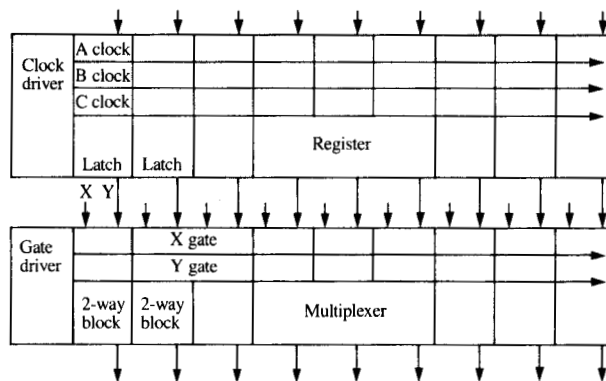


Figure 1 Hierarchical concept of logic macros and use of data flow to simplify wiring.

These basic concepts were followed in the design of our chip physical structures and are fundamental to the development of a chip design methodology that is highly constrained, structured, and repetitive in its use of circuit macros. Nevertheless, the resulting chips still were among the most complex bipolar semiconductor structures ever manufactured at IBM. They combined, for the first time, a level of customization previously found only in FET chips with an advanced bipolar process previously used only in gate-array logic chips and high-speed memory chips. Because of this complexity, and because multiple design passes in VLSI are prohibitively expensive and time-consuming, we believed a physical design verification methodology that guaranteed first-pass success was a necessity. This methodology is described in detail elsewhere in this issue in the paper by McCabe and Muszynski [7].

Technology

The four chips were manufactured by a semiconductor process identical to the one developed for the random-logic chips contained in the IBM 4300 computer series [2], but with horizontal ground rules based on smaller image sizes. This process is characterized by three levels of metal, recessed oxide isolation, and a $2\text{-}\mu\text{m}$ -thick epitaxial layer.

The choice of a basic logic circuit type was arrived at by a process similar to that described in [8] and, for similar reasons, came down to choosing between T^2L (*transistor-transistor logic*) and D^2L (*diode-diode logic*). D^2L is similar to STL (*Schottky transistor logic*) [9], except that the *low-barrier Schottky diodes* (LBSDs) are at the input of the circuit cell instead of the output. This gives the advantage of a single wire output, rather than multiple outputs at the expense of a slightly larger cell, and it simplifies wiring. Another difference is the addition of a *high-barrier Schottky*

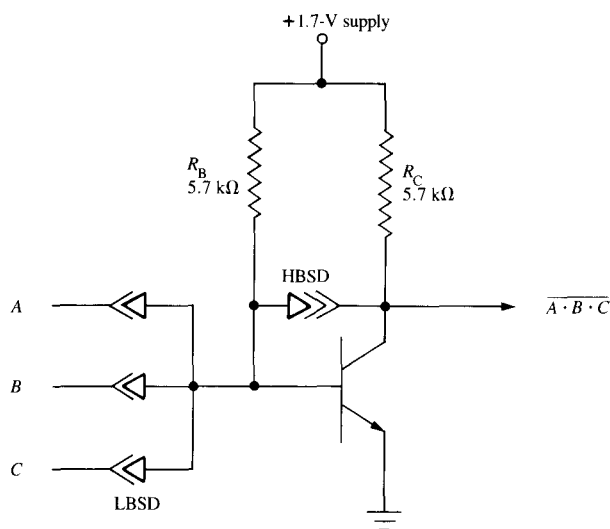


Figure 2 D²L circuit with fan-in of three used in the VLSI microprocessor chips.

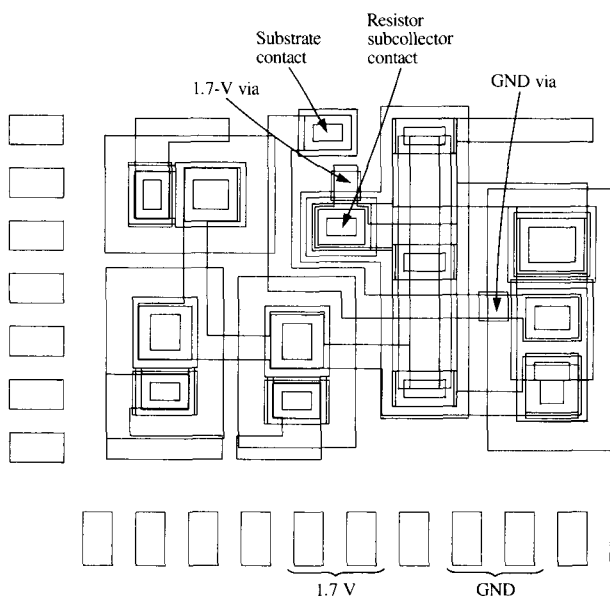


Figure 3 Layout of D²L circuit with fan-in of three.

diode (HBSD) clamp to keep the transistor out of saturation and to improve performance at the expense of some noise tolerance.

The decision was made in favor of the D²L circuit shown in Fig. 2. The speed-power product of this circuit is almost identical to the T²L circuit [8], and the reason for the choice was that it met our layout requirements better than the T²L.

Table 1 Logic capability and performance of the D²L circuit.

Logic capability	Performance
Maximum fan-out (FO) = 6	(1) Calculated average delay = 1.72 ns (nominal) with
Maximum collector dots = 8	FO = 1, FI = 3, wire capacitance $C_w = 0.4$ pF
Maximum fan-in (FI) = 5	Tolerance on delay = $\pm 60\%$
	Nominal power dissipation = 0.52 mW
	(2) Measured based on recirculating loop with FO = 1, FI = 3, $C_w = 0.45$ pF
	(a) East Fishkill wafer (4 sites): average delay = 1.76 ns
	(b) East Fishkill wafer (6 sites): average delay = 1.59 ns

Table 1 summarizes D²L logic capability and performance for the circuit of Fig. 2. Calculated values were obtained by the circuit simulation program, ASTAP [10]. Measured values are from test-site chips containing recirculating loops which were included on the product wafers.

A layout corresponding to the circuit of Fig. 2 is shown in Fig. 3. All building blocks were designed to be a uniform ten vertical wiring channels in width to conform to the chip design strategy. This typically (fan-in of three) gives four free and clear wiring channels (not connected to a block logic service terminal in the second-level-metal data flow. Power distribution metal is in two channels (1.7 V and GND) and uses up four signal wiring channel positions. Extending the 1.7-V and GND channels vertically, one can see that they pass directly over the 1.7-V and GND second-to-first-level vias. In addition, note that a first-level GND connection runs to a substrate contact and the 1.7 V is connected to the resistor subcollector diffusion. Height is variable and increases with higher fan-in (up to five). The three-way AI (AND-INVERT) block shown in Fig. 3 is 10×7 wiring channels or $90 \times 63 \mu\text{m}$ in cell size. Minimum dc noise tolerance for the circuit of Fig. 2 is 130 mV. In the design of the chips, the dc noise tolerance was apportioned as follows:

Ground shift	60 mV
Signal line loss	35 mV
Noise margin	35 mV
Total:	130 mV

The ac noise tolerance always exceeds the dc noise tolerance and becomes infinite in the limit as the noise pulse width goes to zero. For noise pulse widths as large as 5 ns, the noise

tolerance reduces to essentially dc noise tolerance when circuit speeds are on the order of 1 or 2 ns. A 35-mV noise margin is considered adequate for pulse widths this large.

A signal line loss of 35 mV translates to a line length of 2500 μm under maximum loading conditions (cluster at far end), or only about a third of the distance across the chip. We found that a large number of nets exceeded this length. In addition, a maximum fan-out of six was insufficient in many cases. This led to the development of two variations of the fundamental circuit (AI-1M) with no impact on its layout. The differences between the three circuits are tabulated in Table 2. The test configuration analyzed by ASTAP to determine minimum dc noise tolerance is shown in Fig. 4. The corresponding transfer curves from which the minimum dc noise tolerances were taken are shown in Fig. 5. Since the AI-1M circuit has less noise tolerance than the other two types, it was always used as the first circuit in the pair of two circuits in Fig. 4 to generate in-phase gain curves. The configuration was then analyzed for each of the three types as the circuit under test, with loading at maximum, according to Table 2. Process parameters were allowed to vary according to their specified distributions, and the minimum dc noise tolerance was determined by the worst case out of 1000 cases.

Other fundamental logic circuits used in the VLSI microprocessor included a D²L circuit with a larger transistor for driving 18 loads, an exclusive-or (XOR) [11], and the glitchless latch (G-latch). All are comparable on a per-stage basis to the D²L circuits in Table 2 in terms of power, performance, and noise tolerance. The XOR and the G-latch, however, represent radical departures in circuit operation.

The XOR and G-latch circuits shown in Figs. 6 and 7, respectively, are examples of how special functions were efficiently implemented in our pseudo-custom design approach [12]. The XOR, if implemented in unit logic,

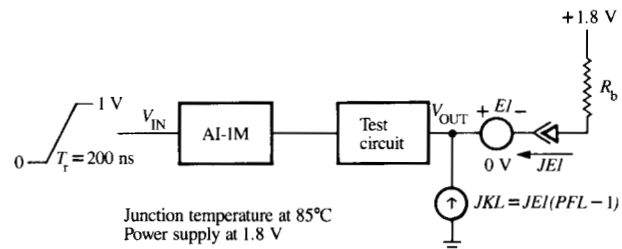


Figure 4 Test configuration for minimum noise tolerance. Legend: JEI = Current from one load; EI = Dummy voltage source used to monitor JEI ; PFL = Maximum number of loads; JKL = Total load current minus current from one load; T_r = Rise time of V_{IN} waveform.

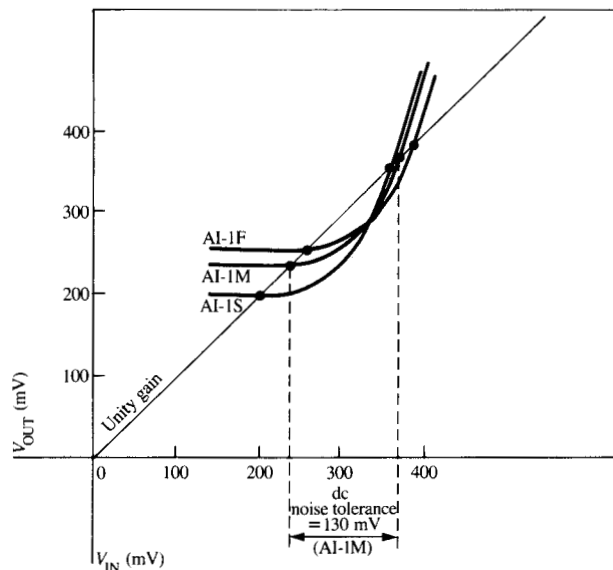


Figure 5 Down-level transfer curves for three variations of the D²L circuit.

Table 2 Variations of D²L circuit using the same transistor device.

Type	HBSD	R_B (k Ω)	R_C (k Ω)	Perf. $FO = 3$ (ns)	Power dissipation (mW)	Max. no. of loads	Min. dc noise tol. (mV)	Max. line length (μm)
AI-1M (Fig. 2)	yes	5.7	5.7	1.43	0.52	6	130	2500
AI-1F	yes	5.4	3.4	1.34	0.68	4	170	8000
AI-1S	no	5.4	3.4	1.96	0.68	9	160	3000

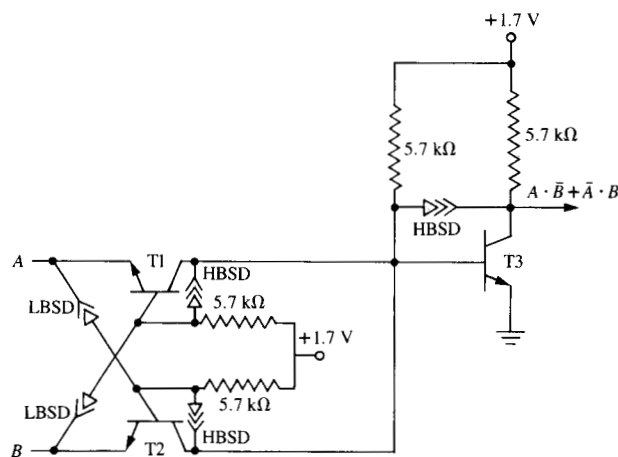


Figure 6 XOR logic circuit.

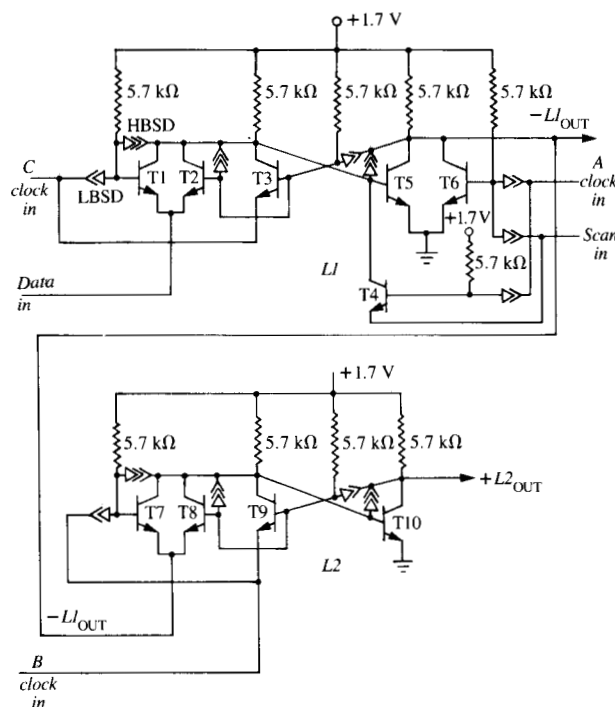


Figure 7 G-latch circuit.

would require four cells to provide an in-phase function, whereas the circuit of Fig. 6 was constructed in a $90 \times 90\text{-}\mu\text{m}$ cell, less than the area of two unit-logic cells. Its operation can easily be understood by inspection. If A and B are of opposite polarity, either $T1$ or $T2$ will conduct and $T3$ will turn off, giving an up level or "1" at the output.

The G-latch includes both $L1$ and $L2$ of a *level-sensitive scan design* [13] latch. The STL version requires seven unit-logic cells to implement this function. The G-latch layout is $90 \times 225\text{ }\mu\text{m}$, about the same as four unit-logic cells. Furthermore, only positive clock signals are used, thereby reducing the number of clock inputs from six to three, which greatly improves wireability on the chip. Finally, the operation is free of the glitch associated with the skew between positive and negative clock pulses. In the $L1$ part, $T3$ and $T5$ are the latching devices. A positive C clock pulse unlatches the cross-coupled pair and allows data to be entered through $T1$. $T2$ holds the collector of $T3$ negative to prevent a glitch when a negative data signal has been entered and the C clock drops. Both the A clock and B clock operate on the latch in a similar fashion.

A PLA circuit description is given in [14]. PLAs were algorithmically generated according to application requirements. However, only three types, each with a different set of resistor values to vary power and performance, were used. Within the three types there is further variation on size and personalization. There are 49 unique PLAs used on the four chips. A typical example has 24 product terms, 17 ns delay, power dissipation of 55 mW, and an area of 0.492 mm^2 .

Search and read lines in the PLAs are constructed by long subcollector diffusions. First-level metal orthogonal to the subcollector diffusions is used for the product term lines. Wherever a contact is placed at a crosspoint, an HBSD, the array element, is created. Wiring channels exist between the search and read lines on the second level, giving a porosity to the PLAs for data flow passing through. The PLA macros were assembled and personalized automatically by software [5] that operates on a logic designer's input and creates a physical design from physical cells and rules stored in its data base.

The 50K-bit ROS is described in detail in [15]. Its array is constructed in the same way as the PLA arrays. The average access time is 24 ns and the average power dissipation is 520 mW. It occupies an area of $2.53\text{ mm} \times 6.37\text{ mm}$, less than half of one chip.

Off-chip drivers are identical to those described in [9]. However, the off-chip receiver of [9] has too low an input impedance for our application, and we chose the circuit of Fig. 8, which is essentially a high-input-impedance emitter-follower driving an inverter. In all of the logic circuits, the option to remove the HBSD clamp exists when additional noise tolerance at a lower performance is desired.

Chip physical structure

The size, density, and complexity of the VLSI microprocessor chips required a coordinated, highly interactive design

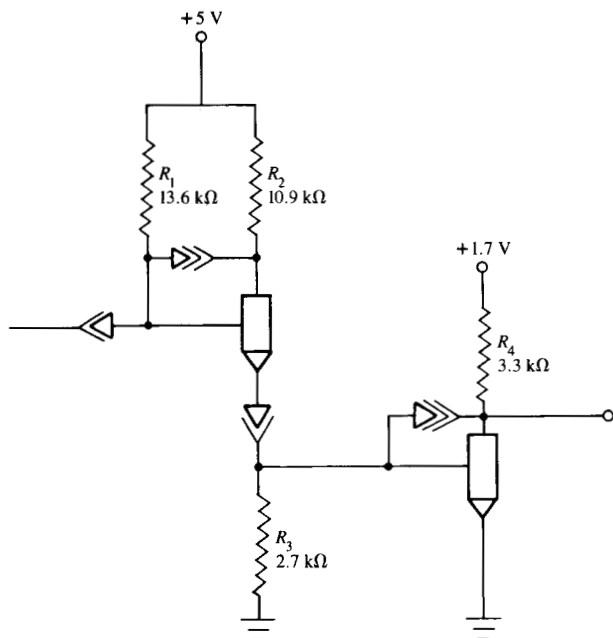


Figure 8 Off-chip receiver.

effort among the circuit, chip, and logic designers in order to meet schedule and optimize chip size to reduce cost. Fundamental to the design of the chip physical structure are two constraints suggested by Fig. 1. These are a common horizontal pitch for building blocks with fixed wiring channels, and maintaining the control lines and data flow on separate levels of wiring. An iterative process took place in which the circuit designer worked with the logic designer to determine types and variations of circuits, and also worked with the chip designer to make the circuits' physical structures conform to the chip's physical structure. The circuit designer then minimized the horizontal pitch of the circuit building blocks in a manner consistent with good wire-through capability. This resulted in the basic D²L logic structure shown in Fig. 3, and a chip wiring structure as illustrated in Fig. 9, which shows macro shadows, first- and second-level signal wiring, and second-level power buses (wide vertical metal). Implicit in the resulting structure is the fact that minimum horizontal ground rules on image sizes and reliability considerations on current-carrying capability had to be satisfied. Figure 9 should be compared with Fig. 1 to compare the implementation to the idealized concept, and with Fig. 3 to understand how the layout of an individual circuit fits into the chip wiring structure.

Third-level metal is used primarily to distribute power over the whole chip with minimum resistive and inductive loss. It is also used to make I/O pad connections. It is well

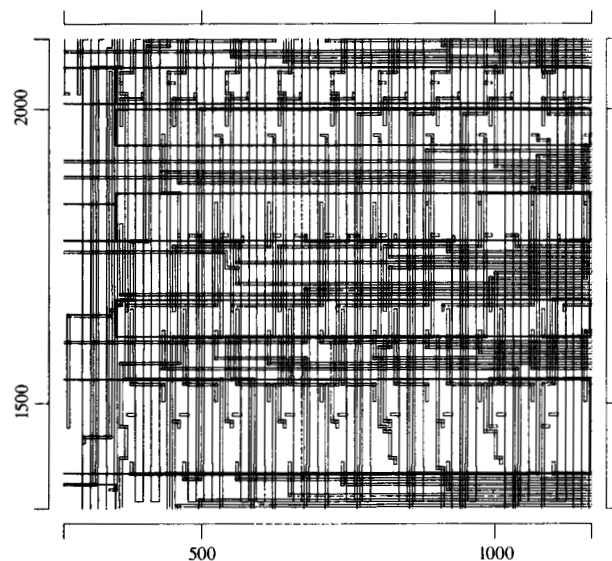


Figure 9 Chip wiring structure of physical macros.

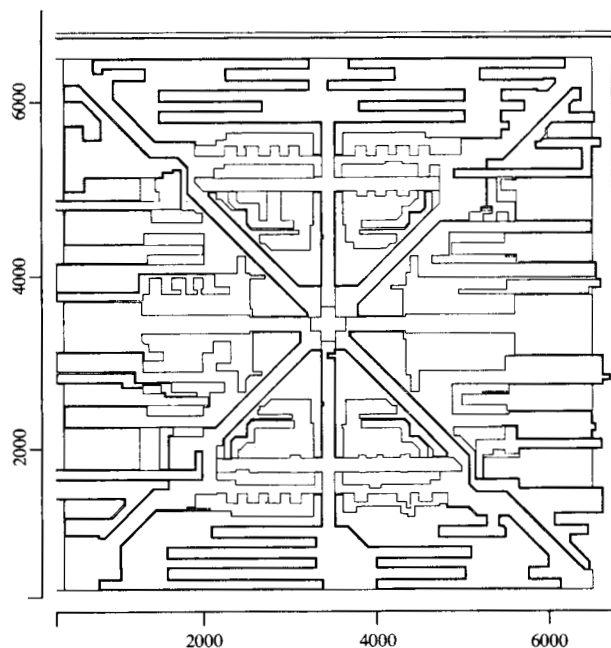


Figure 10 Third-level-metal personalization.

suited to this since it has the largest image size ground rules, and these are consistent with the wide conductors required. A third-level-metal personalization appears in Fig. 10.

Data flow wiring is on second-level metal to keep it free and clear of the internal circuit connections. Control lines

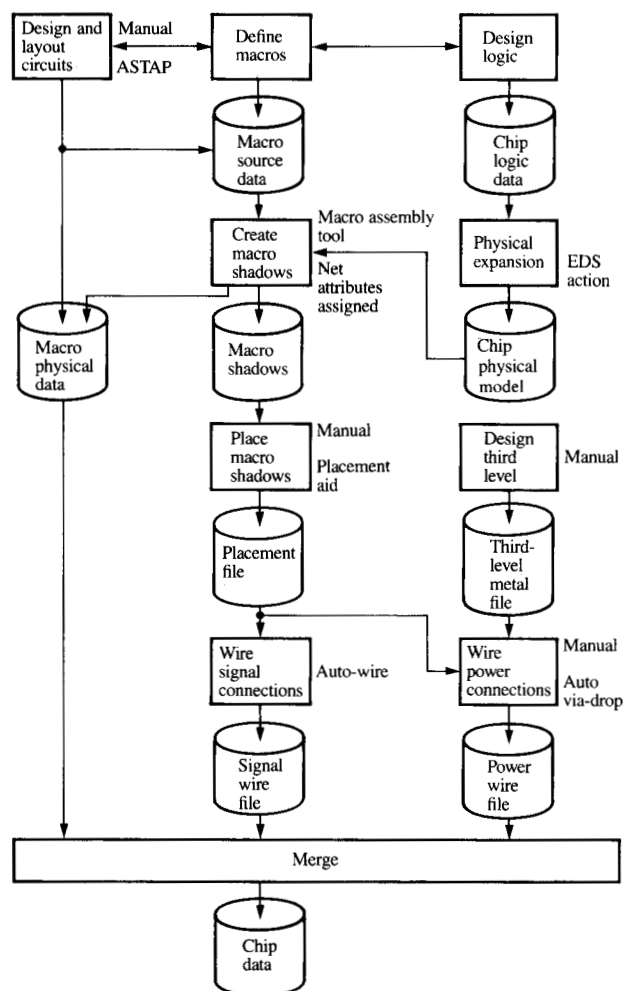


Figure 11 Chip physical design flow.

such as gate lines or clock lines are on first-level metal because they can be more easily designed into the internal metal of the macro physical designs. Second-level power distribution runs parallel to the data flow and supplies power to every circuit on the chip. Figure 10 shows that third-level power distribution is orthogonal, at the top and bottom of the chip, to the second-level-metal power distribution as shown in Fig. 9. This facilitates power-via connections between these levels, although via connections are also made on diagonal crossings wherever a potential via site exists. An important signal wireability consideration is that first-to-second-level vias can be placed at any available location. To maximize availability, we needed adjacent via capability. Therefore, the via image ground rules became the determining factor in defining our wiring pitch, which is the same for first and second levels. Because it is the same, we have a further freedom of rotating a macro 90° to the normal direction of data flow, if this is necessary, without causing ground rule violations and difficulties in wiring the chip.

Chip physical design methodology

The development of the chip physical design methodology was driven by the need to improve productivity, guarantee quality, reduce computer expenditure, and generally make the design task more manageable. This was accomplished by designing each chip with macro shadows rather than with the complete set of physical data for each macro. These shadows contain only the physical data required to wire the chip. These data, plus rules from the circuit designer such as minimum allowable power supply voltages at the circuit and maximum signal wire lengths, are sufficient to allow the chip designer to complete the chip design to a point where it can be verified both logically and electrically to the macro boundary level. Macro design is verified prior to chip design verification. The physical design verification aspects are fully discussed in detail elsewhere in this issue in the paper by McCabe and Muszynski [7].

The design flow is shown in Fig. 11. In the early stages of the design, much interaction takes place between the logic designers and physical designers to coordinate the logic design and the circuit design, and to define the macros which are necessary to perform the logic. Macros assembled from building blocks are not manually physically designed. Their physical design is done by a macro assembly software tool. The source data for this software tool are derived from the circuit layouts and define the building blocks and macros with respect to perimeter, wiring channel blockage, and terminal locations. All dimensions are in terms of wiring channels. The macro assembly software tool creates shadow cells for all the building blocks and macros containing lead attributes necessary for macro design verification. The shadow information is then added to the layout information in the *macro physical data* file so that the macros may be design-verified.

The logic design of the chip is accomplished according to rules from the circuit designer on fan-in, dotting, and loading. As illustrated in Fig. 11, this design is physically expanded by an Engineering Design System (EDS) subroutine to provide a chip physical model. EDS is a software package developed for IBM's internal use as a general design aid [6]. The chip physical model contains all of the macro block names and associated net attributes which denote how the blocks are interconnected. Another software tool produces net-attributed macro shadows from the shadows created by the macro assembly software tool, and the chip physical model data. These shadows are used for placement.

Placement is a critical activity and affects wireability, performance, power distribution, and noise tolerance. Electrical integrity is the first priority in chip design, and the placements of the drivers affect this more than any other blocks. Studies performed early in the program indicated

that drivers must be placed directly over the pad array to avoid ground distribution losses, simultaneous switching noise in the power distribution, and losses in the output lines to the pads. The logic macros must be placed so that interconnection lengths in general and critical paths in particular are kept as short as possible.

The chip designer enlists the aid of the logic designer in performing a placement. Together they determine the wiring affinity between macros and the critical paths with the help of the machine logic diagrams. In addition, a placement aid is used which assesses the wireability of a placement and also indicates the best placement for a particular macro on the basis of wireability only. Figure 12 shows the shadow placement of a VLSI microprocessor chip and gives an example of how the placement aid is used. The placement aid assesses the wireability of the placement by projecting the percent of wiring channel capacity used and comparing it to the number of wiring channels remaining. In the example shown, this is done by scanning the vertical channels horizontally and plotting the results on the scales to the right. The light line represents percent capacity used. Note that it goes from zero on either end of the chip to a maximum in the center. The heavy line represents the number of empty wiring channels remaining. Where the capacity used is a maximum (light line) and the capacity remaining is a minimum (heavy line), wireability will tend to be poor, and the placement is adjusted to alleviate this condition as much as possible.

The contours on Fig. 12 represent lines of equal but increasing wire length associated with a particular macro as its placement is moved further from the center of the contours; the center is optimum for short wire length. Both of these features are shown here for vertical wiring only. A similar plot would be obtained for horizontal wiring to complete the wiring information.

Once the placement is established, two separate wiring activities take place. Signal wiring is done by the wiring program and power connections are made from the third-level metal to the macros. The wiring program operates on the placement file in Fig. 11, which contains net-attributed shadows and their coordinate locations on the chip. The program recognizes first- and second-level wiring blockage, and the net attributes at the terminal locations of each shadow. It connects all the terminals having the same attributes, using the shortest possible paths. All vertical wiring was done on the second level for the VLSI microprocessor chips. Although the program wires orthogonally on separate levels, the choice of level is arbitrary to the program. If not successful the first time, the program will iterate by moving blocking wires, and sometimes completes all wires in this fashion. After the wiring pass is completed, a wireability analysis similar to the placement projection is made and the

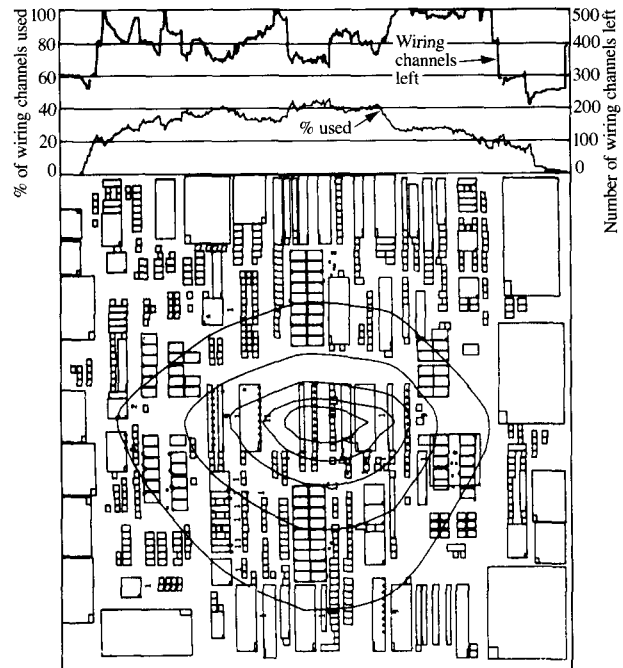


Figure 12 Shadow placement of Chip 2 and use of the placement aid.

designer has the option of manually trying to imbed any remaining overflows or changing the placement and rerunning the program.

Via connections from the predesigned third-level structure to the second level are greatly facilitated by an automatic via-dropping program, which takes advantage of the many crosspoints between the regular vertical structure of second-level power distribution and the third-level power distribution of the same voltage or ground. The via-dropping program eliminates much manual work, puts in vias correctly as per image ground rules, and results in an optimum use of power distribution vias since they are placed in every possible via site. This further reduces power distribution losses and contributes to a good electrical design.

The macro physical data, the signal wire data, and the power wire data all exist as separate data sets and are design-verified separately [7] before being merged as the final step in Fig. 11. Additional verification is performed on the merged data; this is also described in greater detail in [7].

The methodology described here allowed us to accomplish each chip design with two persons and assistance from a chip logic designer. A six-month chip design cycle occurred in

Table 3 VLSI microprocessor chip statistics.

Chip no.	Function	Chip edge size (mm)	No. of logic macro blocks	No. of PLAs	ROS	No. of devices	No. of equiv. circuits	I/O	Power (W)
1	ALU CLK MACH CHK	7.2	895	10	no	24493	4400	143	3.5
2	INST DEC, INTRPT, ROAR	6.8	702	15	no	22843	4297	153	3.5
3	MEM ADDR MEM CTRL I/O CTRL	6.8	479	16	no	22235	4357	153	3.8
4	RAM CTRL RAM ADDR	6.8	441	8	50K	33087	2502 (ROS not incl.)	148	3.2
Totals:			2517	49	50K	102658	15556	597	14.0

each of two passes, but in the first pass some of this time was due to the methodology still being developed based on what we had learned from an experimental learning vehicle chip [14]. In the second pass, some of the time was given to logic design changes. Given the complete methodology and a fixed logic design, three to four months would be adequate to design a chip. Most of the project development time and resources were spent in developing the macros and the methodology, including the general chip structure and design verification methodology [7]. Over a three-year period, from 6 to approximately 20 people, mostly engineers, worked on this aspect of the project. Each pass resulted in 100 percent functional chips with the exception of one of the four part numbers in the second pass. This part had a design error in the wiring which was found by our checking programs but was overlooked due to human error. The achievement of our excellent results in error-free designs is a tribute to the physical design methodology [7].

Discussion of chip statistics

A summary of chip statistics for the four VLSI microprocessor chips is given in Table 3. These chips are the most complex bipolar product chips ever developed in IBM up to the time of this writing. The number of equivalent circuits was arrived at for each chip by reducing the logic in integrated functions to AI (AND-INVERT) circuits with fan-in ≤ 3 . This normalizes the count to a gate array chip

circuit count, where the gate cells have a maximum fan-in of 3. The total circuit count of 15 556 circuits contained in a 50 $\times$ 50-mm module implies the highest bipolar density at the module level ever achieved to date by IBM, with 6.2 circuits per square millimeter of module substrate area.

Distribution of net lengths is an important factor in chip design. Short net lengths imply better performance, improved noise tolerance, and more wireability. A major objective of our chip design strategy was to minimize net lengths by developing an orderly chip structure and by relating the physical design to the logic design. Global net statistics (nets external to macros) were automatically generated by the wiring program. However, because many of the nets that would have been external to circuit cells in a gate array are integrated inside of macros, we had to count these nets manually on the basis of connections between AI circuits with fan-in ≤ 3 . These statistics are plotted in Fig. 13 for Chip 2.

Figure 13 shows that most of the nets (71.3 percent) on the chip are internal to the macros. These nets are all under 1 mm in length and many are much less than that. Another 400 nets from the global category fall into this net-length range. Altogether, 82.6 percent of the nets are 1 mm or less in length and, in terms of capacitance, are about 0.5 pF or less. Delay sensitivity to capacitance is highest for the AI-1M

circuit driving one load, with a turn-off (rising) transition at 1 ns per pF. Therefore, delay degradation is generally less than 0.5 ns for about four out of five nets on the chip. A small percentage of nets exceed the 8-mm limit shown in Table 2, and one net exceeded 20 mm. These nets present a problem not in performance, because the chip designer minimizes critical path net lengths, but rather in noise tolerance. This is because the voltage drop from the load circuit to the driving circuit due to load current and line resistance degrades the down level at the input to the load circuit. Each of these exceptions was analyzed and a judgment was made that it could be accepted on the basis of either reduced circuit loading on the net or reduced power distribution loss (from that assumed for the rule).

Another interesting point about the relatively small proportion (28.7 percent) of nets that were globally wired is the reduced load on the automatic wiring program. Similar statistics exist on all four chips, indicating that a large part of the burden of wiring a VLSI chip has been absorbed by the macros.

Summary

The engineering complexity of the VLSI microprocessor bipolar chip design task forced us to plan an approach that would allow us to proceed in an orderly fashion with limited resources and achieve success. A hierarchical logic macro structure was developed which, along with PLAs and a ROS, could be related to the logic design and simplify the chip placement task. Simultaneously, a chip structure was developed to provide an optimized power distribution and an orderly arrangement of signal and power wiring to accommodate the placement of the macros. Placement and wiring of the chips then became a straightforward process, which was facilitated by the use of a software placement aid, an automatic wiring program, and a power-via-dropping program.

The four chips designed by this methodology each exceed 4000 circuits in terms of equivalent unit-logic circuits and/or ROS, and are engineered to operate per specifications over all specified ranges of environment, process parameters, and application. Internal delays are typically less than 3 ns per stage. Several fully functional first-pass machines were built from the four chip part numbers. These machines met all specified machine requirements. A second pass on the design was made to incorporate some logic changes desired by the machine designers.

Acknowledgments

Many people in IBM in Kingston and East Fishkill, NY, contributed to this effort. The authors wish to acknowledge the following individuals. In Kingston, the logic circuits were designed and, in some cases, invented by H. Askin. R. Hatch

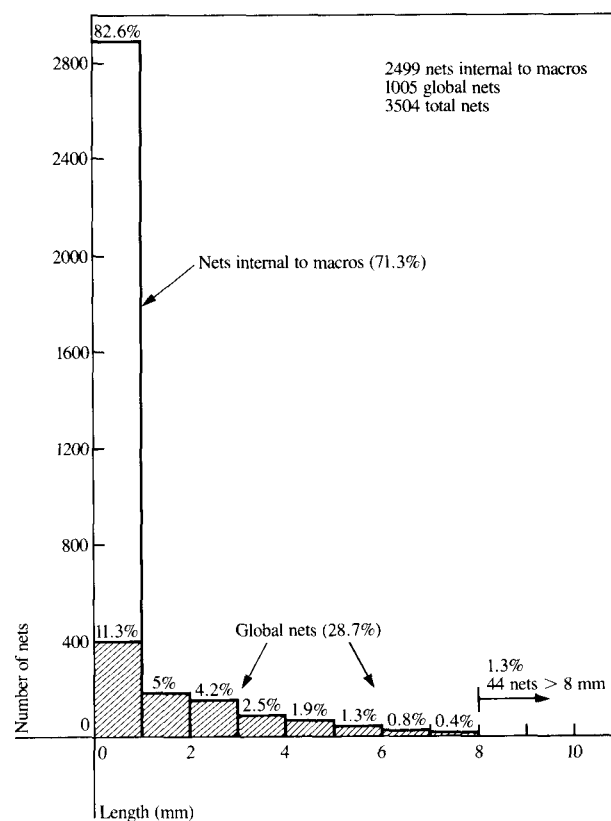


Figure 13 A histogram of net lengths for Chip 2.

and J. Ludwig were responsible for the PLAs and ROS, respectively. R. Proctor developed much of the graphic methodology and J. McCabe developed the overall design methodology. J. Coleman and S. Bello made important management contributions to accomplishing the layout and wiring of the four part numbers on schedule. In East Fishkill, Y. Ting provided the product engineering interface; D. Gittleman, T. Chang, and C. Peterson supplied valuable technical help and facilitated the manufacturing of the chips at that site; and P. Salvatori tested the chips. J. Wyn Jones of IBM World Trade in Hursley, UK, contributed his expertise in the implementation of PLA macros while on assignment in Kingston. Special thanks are also due E. Eichelberger for his encouragement and management support.

References and note

1. A. J. Blodgett and D. R. Barbour, "Thermal Conduction Module: A High-Performance Multilayer Ceramic Package," *IBM J. Res. Develop.* **26**, 30-36 (1982).
2. J. Pomeranz, R. Nijhaus, and C. Vicary, "Customized Metal Layers Vary Standard Gate-Array Chip," *Electronics* **52**, 105-108 (1979).

3. John E. Campbell and Joseph Tahmoush, "Design Considerations for a VLSI Microprocessor," *IBM J. Res. Develop.* **26**, 454-463 (1982, this issue).
4. Joseph C. Logue, Walter J. Kleinfelder, Paul Lowy, J. Randal Moulic, and Wei-Wha Wu, "Techniques for Improving Engineering Productivity of VLSI Designs," *IBM J. Res. Develop.* **25**, 107-115 (1981).
5. R. L. Golden, P. A. Latus, and P. Lowy, "Design Automation and the Programmable Logic Array Macro," *IBM J. Res. Develop.* **24**, 23-31 (1980).
6. P. W. Case, M. Correia, W. Gianopoulos, W. R. Heller, H. Ofek, T. C. Raymond, R. L. Simek, and C. B. Stieglitz, "Design Automation in IBM," *IBM J. Res. Develop.* **25**, 631-646 (1981).
7. J. F. McCabe and A. Z. Muszynski, "A Bipolar VLSI Custom Macro Physical Design Verification Strategy," *IBM J. Res. Develop.* **26**, 485-496 (1982, this issue).
8. A. H. Dansky, "Bipolar Circuit Design for a 5000-Circuit VLSI Gate Array," *IBM J. Res. Develop.* **25**, 116-125 (1981).
9. J. Z. Chen, W. B. Chin, T.-S. Jen, and J. Hutt, "A High-Density Bipolar Logic Masterslice for Small Systems," *IBM J. Res. Develop.* **25**, 142-151 (1981).
10. *Advanced Statistical Analysis Program (ASTAP)*, Program Reference Manual, Order No. SH20-1118-0; available through IBM branch offices.
11. H. O. Askin and J. Kau, "Buffered Exclusive-or," *IBM Tech. Disclosure Bull.* **25**, No. 3a, 1028-1029 (1982).
12. For examples of other solutions to the glitch problem, see R. L. Hart and J. C. Leininger, "Latch Circuit," U.S. Patent 3,740,590, June 19, 1973; E. Berndlmaier, J. A. Dorler, and U. Olderdissen, "Glitchless T²L Latch Utilizing Single-Phase Clock Input," *IBM Tech. Disclosure Bull.* **18**, 1404-1405 (1975); and E. B. Eichelberger and G. J. Robbins, "High Performance Latch Circuit," U.S. Patent 3,986,057, Oct. 12, 1976.
13. E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability," *Proceedings of the 14th Design Automation Conference*, New Orleans, LA, 1977, pp. 462-468.
14. K. F. Mathews, J. J. Coleman, and Y.-M. Ting, "Custom Macro Design in VLSI Bipolar Technology," *International Solid-State Circuits Conference Digest of Technical Papers* **25**, 56-57 (1982).
15. J. A. Ludwig, "A 50K Bit Schottky Cell Bipolar Read-Only Memory," *IEEE J. Solid-State Circuits* **SC-15**, 816-820 (1980).

Received October 22, 1981; revised February 8, 1982

K. F. Mathews is located at the IBM General Technology Division laboratory, East Fishkill Facility, Hopewell Junction, New York 12533; and L. P. Lee is located at the IBM System Products Division laboratory, Neighborhood Road, Kingston, New York 12401.